MEDIUL LIMBAJULUI DE PROGRAMARE STUDIAT

Prin codificarea algoritmului în limbajul de programare ales, obţinem un program. Programul este transmis calculatorului prin introducerea textului de la tastatură (editare) şi memorat într-un fişier sursă cu extensia pas pentru limbajul Pascal (exemplu: program1.cpp) sau extensia cpp pentru limbajul C/C++ (exemplu: program1.cpp). […]

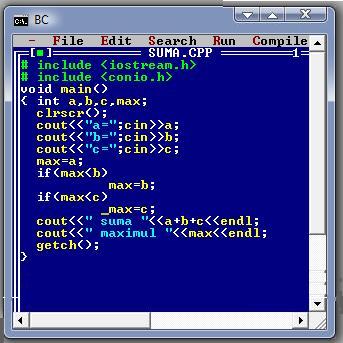
„Traducerea” textului de program din limbajul de programare ales în limbajul intern al calculatorului se face de către compilator prin compilare. Compilatorul realizează şi verificarea sintactică a programului, semnalând erorile detectate. Corectarea erorilor se face prin editare, de la tastatură, fiind urmată de o nouă compilare. Când textul programului nu mai conţine nicio eroare, compilatorul generează un fişier obiect cu extensia obj (exemplu: program1.obj). Fişierul obiect este executat prin comanda Run.

Pentru a oferi programatorilor accesul la fişiere şi la resursele de editare, compilare şi execuţie, au fost dezvoltate mediile de programare.[…]

Datele cu care lucrează un program pot fi memorate în fişierele text. Astfel se asigură păstrarea datelor şi reducerea timpului de operare la testarea/utilizarea programului.

Un fişier text este format dintr-o succesiune de caractere ASCII scrise pe una sau mai multe linii (rânduri), pe memorie externă (disc). Sfârşitul fişierului este marcat de o etichetă specială: EOF (End Of File). Prelucrarea unui fişier se face prin mai multe operaţii: deschidere (pentru citire sau scriere), citire, scriere, închidere. […]

Editarea liniilor de program se face de la tastatură.



Suntem în situaţia de a alege unul dintre mai mulţi algoritmi care rezolvă aceeaşi problemă. Cum alegem cel mai bun algoritm? După ce criteriu?

Estimăm timpul de execuţie. Programatorul îşi pune problema determinării timpului de execuţie chiar în faza proiectării algoritmului. Este vorba de un timp teoretic întrucât, practic, timpul de execuţie depinde de viteza de lucru a procesorului.

Timpul teoretic în care un algoritm rezolvă problema depinde de numărul de operaţii executate.

Pentru a compara între ei doi algoritmi, se determină operaţiile de bază, al căror număr de execuţii variază de la un algoritm la altul. În general, acestea sunt operaţii de atribuire şi de compilare. Contează şi numărul de valori (n) cărora li se aplică aceste operaţii.[…]

Numărul de execuţii ale operaţiilor de bază se pot exprima în forma O(n), O(n2), O(n3), formă numită şi ordinul de mărime sau complexitatea algoritmului. Se consideră ca fiind cel mai bun algoritmul cu ordinul de mărime cel mai mic.[…]

Probleme propuse:

1. Se introduc n+m numere naturale.
   1. Să se formeze mulţimea A din primele n valori şi mulţimea B din următoarele m valori.
   2. Să se verifice dacă o valoare oarecare x, introdusă de la tastatură, aparţine mulţimii A.
   3. Să se formeze mulţimea R ca reuniune dintre mulţimile A şi B.
2. Doi elevi lucrează împreună la un proiect pentru ora de istorie. Fiecare s-a documentat şi a întocmit o listă cu cele mai importante evenimente istorice. În proiect, trebuie prezentat o singură dată fiecare eveniment istoric, în ordine cronologică.
   1. Întocmiţi cele două liste cu cel mult 10 evenimente istorice.
   2. Verificaţi dacă cele două liste pot fi interclasate.
   3. Obţineţi lista finală prin interclasarea celor două liste.
   4. Să se determine dacă un an oarecare, A, aparţine listei de evenimente.
   5. Realizaţi un program care să rezolve cerinţele de la punctele a,b,c şi d.
3. Se cunosc rezultatele la Olimpiada de informatică pentru trei şcoli. Numărul de participanţi din fiecare şcoală este n1,n2, respectiv n3.
   1. Propuneţi un algoritm pentru obţinerea clasamentului fiecărei şcoli.
   2. Propuneţi un algoritm prin care să se obţină clasamentul pentru toate şcolile.
   3. Analizaţi eficienţa algoritmului propus.[…]

Complexitatea algoritmilor:

|  |  |  |
| --- | --- | --- |
| Nr.crt. | Complexitate | Tipul de algoritm |
| 1 | O(n) | algoritm liniar |
| 2 | O(n2) | algoritm pătratic |
| 3 | O(n3) | algoritm cubic |
| 4 | O(2n) | algoritm exponenţial |
| 5 | O(log2n) | algoritm logaritmic |

(Adaptat după *Manualul de Informatică, clasa a X-a*, coordonator Mioara Gheorghe)